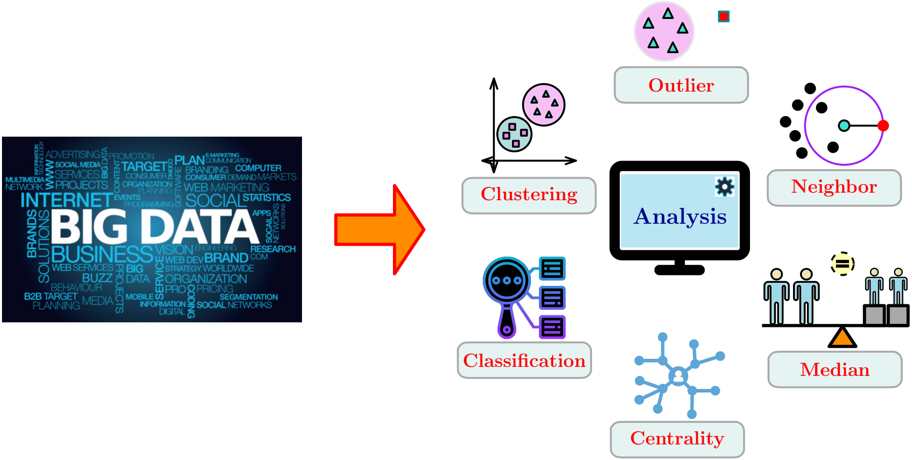
Efficient Data Analytics on Augmented Similarity Triplets

SARWAN ALI

joint work with

I. U. Khan, M Ahmad, U Hassan, M A Khan, S Alam

Big Data Analytics

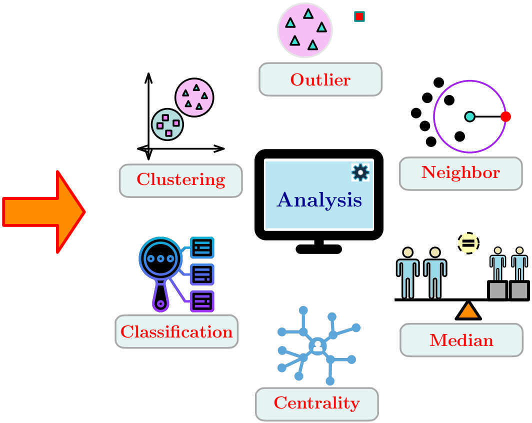


Feature Vector Representation

$$\overbrace{\begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & x_{nm} \end{bmatrix}}^{m \text{ features}}$$

n objects

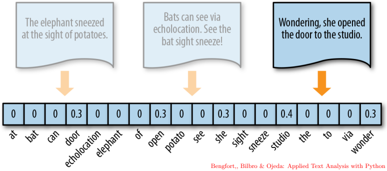
$n \times m$ data matrix



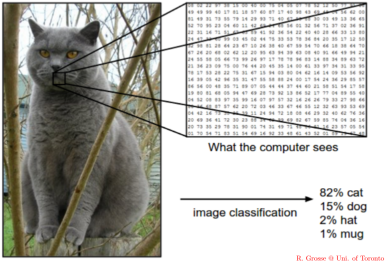
Issues with Explicit Representation

Explicit representation of objects may not be available or meaningful

- No meaningful coordinates for text/image/customer

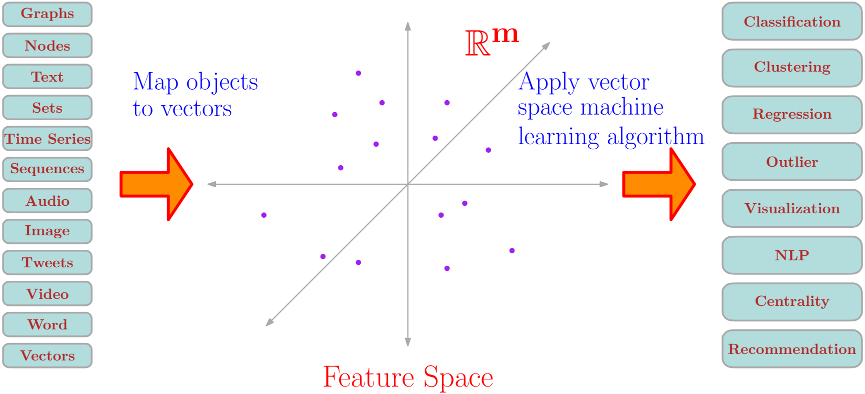


Bengfort,, Bilbro & Ojeda: Applied Text Analysis with Python



R. Grosse @ Uni. of Toronto

Representation Learning



Analytics Require Similarity Measures

Notion of similarity is **sufficient** for data analysis algorithms

- Classification/Clustering: Group "similar" items
- Outlier Detection: Identify items "dissimilar" from others
- Centrality Computation: Evaluate "similarity" of an item to all others
- Nearest Neighbor: Find the most "similar" objects to a query object
- Median: Find the item most "similar" to all others
- Recommendation: Recommend item j to user i if users "similar" to i like items "similar" to j
- Locality Sensitive Hashing: "Similar" items go to same bucket
- Reduce dimensionality: While preserving pairwise "similarities"

Analytics using Similarity

Similarity/Distance Matrix

- Used for Agglomerative clustering, Kernel SVM, Kernel PCA, ...
- Usually computed from explicit representation of objects

m features

$$n \text{ objects} \quad \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & x_{nm} \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & d_{12} & \cdots & \cdots & d_{1n} \\ d_{21} & 0 & \cdots & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & \cdots & 0 \end{bmatrix}$$

$n \times m$ data matrix

$n \times n$ distance matrix

Issues with Proximity Measures



Distance function may not be very meaningful

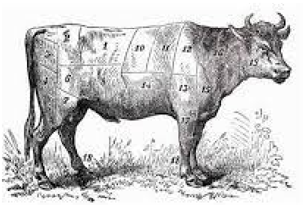
- Which two images are more similar based on shape/purpose?

Issues with Proximity Measures

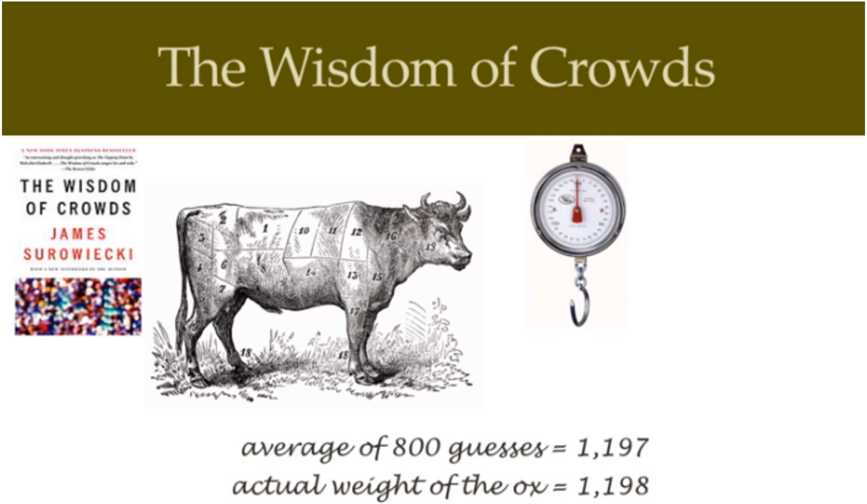
Distance function may not be very meaningful

- Which two images are more similar based on shape/purpose? RGB values of images may not encode perception of images

Human Based Computation



Human Based Computation



Human Based Comparisons

Humans have a hard time to

- Explain embedding coordinate
- Quantify a coordinate value
- Evaluate pairwise similarity $sim(A, B) = ?$

Human Based Comparisons

Humans have a hard time to

- Explain embedding coordinate
- Quantify a coordinate value
- Evaluate pairwise similarity $sim(A, B) =?$

But humans are good at

- Differentiating things perceptually
- Comparing objects' features
- Comparing pairwise similarities $sim(A, B) > sim(A, C)$?

Human Based Comparisons

Humans can easily assess that

Car

Jeep

Truck

A car is more similar to a jeep as compared to a truck, by utility

Human Based Comparisons

Humans can easily assess that

Icecream

Steak

Cookies

Ice cream and cookies are more similar, based on taste

Human Based Comparisons

Humans can easily assess that

Rocky mountains

Snow-coverd peak

Sea-view

Rocky mountains and snow-covered peak are similar, by scenic view

Encoding Comparison Result

Comparison of pairs-wise similarities of three objects encoded as triplets

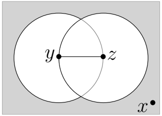
Encoding Comparison Result

Comparison of pairs-wise similarities of three objects encoded as triplets

x is the outlier among the three

Outlier: $(x, y, z)_O$

$$(x, y, z)_O \implies d(x, y) > d(y, z) \text{ AND } d(x, z) > d(y, z)$$



$(x, y, z)_O$

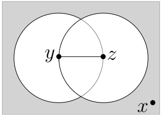
Encoding Comparison Result

Comparison of pairs-wise similarities of three objects encoded as triplets

x is the outlier among the three

Outlier: $(x, y, z)_O$

$(x, y, z)_O \implies d(x, y) > d(y, z)$ AND $d(x, z) > d(y, z)$

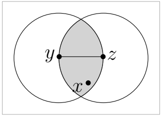


$(x, y, z)_O$

x is the central among the three

Central: $(x, y, z)_C$

$(x, y, z)_C \implies d(x, y) < d(y, z)$ AND $d(x, z) < d(y, z)$



$(x, y, z)_C$

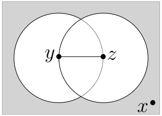
Encoding Comparison Result

Comparison of pairs-wise similarities of three objects encoded as triplets

x **is the outlier among the three**

Outlier: $(x, y, z)_O$

$$(x,y,z)_O \implies d(x,y) > d(y,z) \text{ AND } d(x,z) > d(y,z)$$

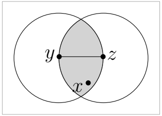


$(x, y, z)_O$

x **is the central among the three**

Central: $(x, y, z)_C$

$$(x,y,z)_C \implies d(x,y) < d(y,z) \text{ AND } d(x,z) < d(y,z)$$

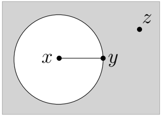


$(x, y, z)_C$

x **is the closer to** y **than** z

Anchor: $(x, y, z)_A$

$$(x,y,z)_A \implies d(x,y) < d(x,z)$$



$(x, y, z)_A$

Convert anything to anchor

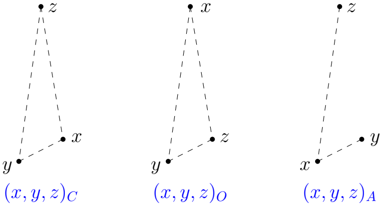
Comparison of pairs-wise similarities of three objects encoded as triplets

Convert anything to anchor

Comparison of pairs-wise similarities of three objects encoded as triplets

Anchor triplet contains the least information

Out of the 3 pairwise distances comparisons, it only provides two



$$(x, y, z)_O \implies (y, x, z)_A \text{ AND } (z, x, y)_A$$

$$(x, y, z)_C \implies (y, z, x)_A \text{ AND } (z, y, x)_A$$

Too many triplets

Since comparisons are easier than computation for humans, triplets are obtained from human sources

Too many triplets

Since comparisons are easier than computation for humans, triplets are obtained from human sources

Distance matrix needs a number of for $\binom{n}{2}$ pairs of objects

The total number of triplets are $\binom{n}{3}$

▷ $n = 300$, $\binom{n}{2} = 44,850$ $\binom{n}{3} = 24,503,050$

Too many triplets

Since comparisons are easier than computation for humans, triplets are obtained from human sources

Distance matrix needs a number of for $\binom{n}{2}$ pairs of objects

The total number of triplets are $\binom{n}{3}$

▷ $n = 300,\ \binom{n}{2} = 44,850\ \binom{n}{3} = 24,503,050$

Statistics to the rescue to avoid getting too many triplets

To estimate a number, no need to measure the whole population or even a percentage of it. A random sample of 1000 can give decent results!

So measure only a small (preferably random) sample of anchor triplets

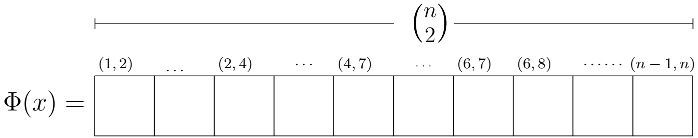
Comparison result as relative ordering

Fix an ordering on objects ▷ $x_1, x_2, \ldots, x_n$

For every object x, consider all triplets with x as anchor

For a pair $x_i, x_j \neq x$, either $(x, x_i, x_j)_A$ or $(x, x_j, x_i)_A$ is possible

$\Phi(x)$ is an $\binom{n}{2}$-dim vector encoding relative ordering of objects w.r.t x



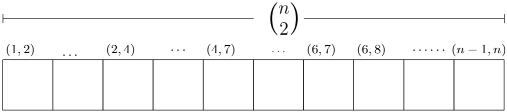
$$\Phi(x)(i,j) = \begin{cases} 1 & \textit{if} \quad (x, x_i, x_j)_A \text{ is a triplet} \\ -1 & \textit{if} \quad (x, x_j, x_i)_A \text{ is a triplet} \\ 0 & \text{else} \end{cases}$$

Feature Vector From Triplets

$\Phi(x_i)$ is an $\binom{n}{2}$-dim vector encoding relative ordering of objects w.r.t x_i

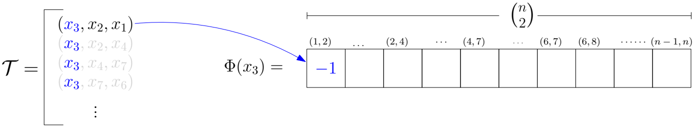
$$\mathcal{T} = \left[\begin{array}{c} (x_3, x_2, x_1) \\ (x_3, x_2, x_4) \\ (x_3, x_4, x_7) \\ (x_3, x_7, x_6) \\ \vdots \end{array}\right.$$

$\Phi(x_3) =$



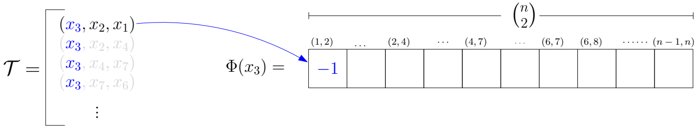
Feature Vector From Triplets

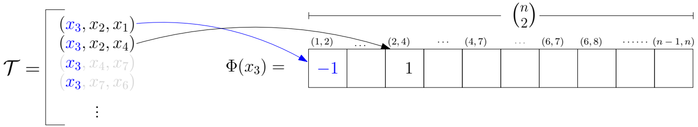
$\Phi(x_i)$ is an $\binom{n}{2}$-dim vector encoding relative ordering of objects w.r.t x_i



Feature Vector From Triplets

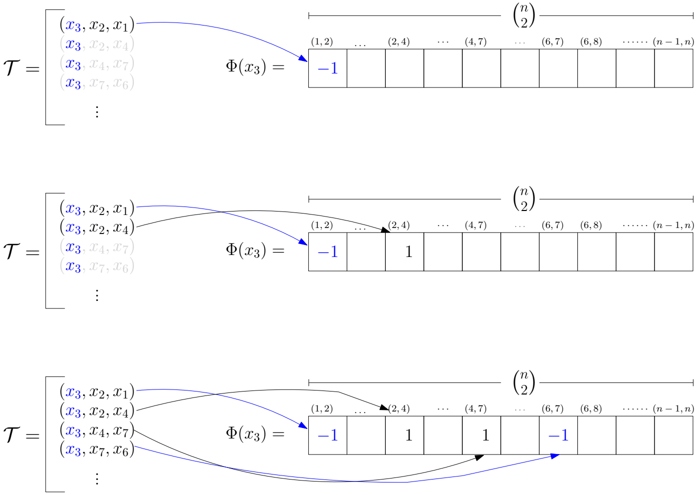
$\Phi(x_i)$ is an $\binom{n}{2}$-dim vector encoding relative ordering of objects w.r.t x_i





Feature Vector From Triplets

$\Phi(x_i)$ is an $\binom{n}{2}$-dim vector encoding relative ordering of objects w.r.t x_i



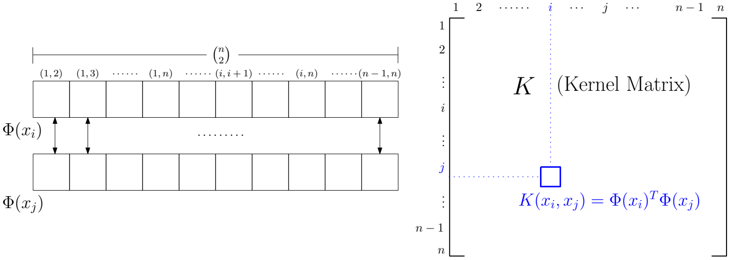
Pairwise Similarity from Triplets

- $\Phi(x)[\cdot] - \Phi(y)[\cdot] = 0 \implies a, b$ ordered the same from x and y
- $\Phi(x)[\cdot] - \Phi(y)[\cdot] = \pm 2 \implies a, b$ ordered differently from x and y
- $\Phi(x)[\cdot] - \Phi(y)[\cdot] = \pm 1 \implies a, b$ ordered from one but not from other

$\Phi(x) \cdot \Phi(y)$ is agreements minus disagreements of pairs orders from x & y

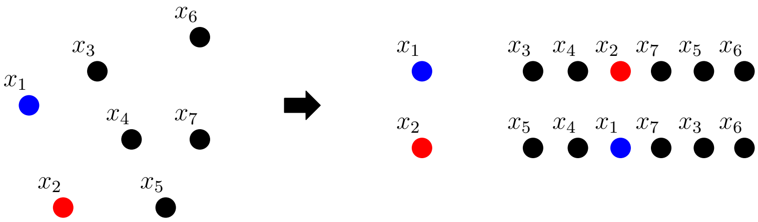
We use this dot product as a kernel ▷ a pairwise similarity measure

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$$



Issues with Kernel

We want a total order on the $n-1$ other objects with respect to an anchor



With limited number of triplets we only get a partial order

Triplets Representations as DAG

- Let $\mathcal{X}$ be the dataset of n objects
- Let $\mathcal{T}$ be the available triplets set

- Represent $\Phi(x)$ as a DAG G_x
- $(x, y, z)_A$ is represented as a directed edge form y to z in G_x
- Formally,

$$E(G_x) := \{(y, z) \mid y, z \in \mathcal{X}, (x, y, z) \in \mathcal{T}\}$$

Triplets Representations as DAG

$\mathcal{T}$

(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)

Directed Graph G_x

Triplets Representations as DAG

$\mathcal{T}$

Directed Graph G_x

(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)

$\mathcal{T}$

Directed Graph G_x

(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)

Triplets Representations as DAG

$\mathcal{T}$ Directed Graph G_x

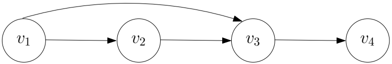
(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)

$\mathcal{T}$ Directed Graph G_x

(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)

$\mathcal{T}$ Directed Graph G_x

(x, v_1, v_2)
(x, v_1, v_3)
(x, v_2, v_3)
(x, v_3, v_4)



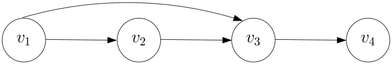
Data Augmentation

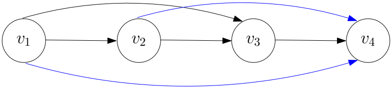
Any reasonable notion of distance/similarity must be transitive

$$d(x,a) < d(x,b) \text{ AND } d(x,b) < d(x,c) \implies d(x,a) < d(x,c)$$

$$(x,a,b)_A \text{ AND } (x,b,c)_A \implies (x,a,c)_A$$

$(x,a,c)_A$ is the extra information extracted form the input





We perform transitive closure on graphs for each object

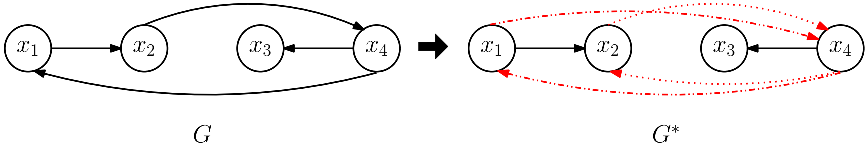
Data augmentation reveals hidden inconsistencies

Human based data is prone to error

An inconsistent pair of triplets

$$(x, y, z)_A \quad \text{AND} \quad (x, z, y)_A$$

can be revealed with data augmentation



Data Analytics from Augmented DAGs

Closeness: $close_x(y)$ is rank of $sim(x, y)$ in decreasing order of $sim(x, \cdot)$

$$close_x(y) = (n-1) - \left|\left\{z \in \mathcal{X}, z \neq x : sim(x,z) < sim(x,y)\right\}\right|$$

We have

- $close_x(y) \;\geq\; deg^{+}_{G_x}(y)$ ▷ lower bound
- $close_x(y) \;\leq\; n - deg^{-}_{G_x}(y)$ ▷ upper bound

Our estimate for $close_x(y)$ is an average of the two bounds

$$close'_x(y) \;=\; \frac{deg^{+}_{G_x}(y) + n - deg^{-}_{G_x}(y)}{2}$$

Data Analytics from Augmented DAGs

Approximate k-nearest neighbors based on estimated closeness

$$k\text{NN}'(x) = \{y \mid close'_x(y) \leq k\}$$

Classification

We use kNN classifier and declare class label of x as the majority among labels of objects in $k'\text{NN}(x)$

k-nearest neighbor graph, kNNG is a graph on vertex set $\mathcal{X}$ such that x is adjacent to k vertices in $k\text{NN}'(x)$

Clustering

We construct kNNG and perform spectral clustering to get clustering $\mathcal{X}$

Experimental Evaluation

We evaluate the quality of our algorithms by appropriate comparison with analytics based on the true similarity matrix of $\mathcal{X}$, $\mathcal{S}(i,j)$.

The following metrics are used

- Kernel Matrix K: To what extent K *agrees* with $\mathcal{S}$ and how well K maintains the order of objects with respect to $\mathcal{S}$
- Centrality and Median: Demonstrate quality of approximate centrality by showing rank correlation between true and approximate centralities
- Nearest Neighbors: Compare true and approximate nearest neighbors
- Clustering: Performing spectral clustering on the nearest neighborhood graph and reporting purity
- Classification: Using the kNN classifier with train-test split of $70-30\%$ to perform supervised analysis

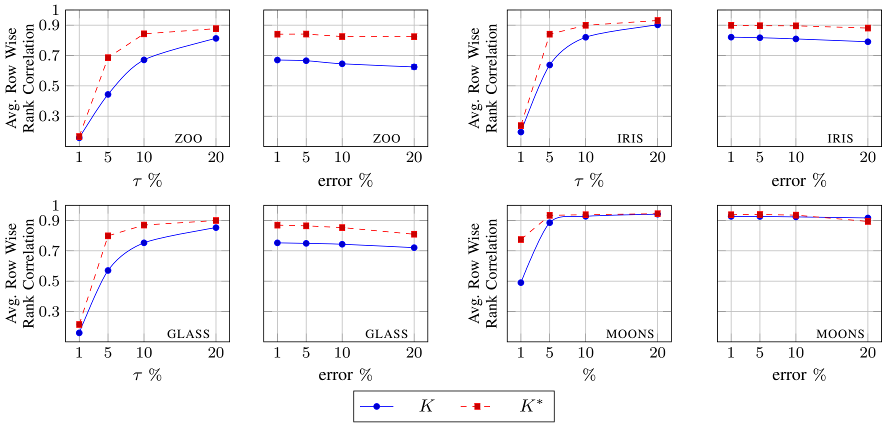
Dataset Description (Real-World)

- ZOO dataset consists of 16-dimensional feature vectors of 101 animals. The dataset has 7 different classes
- IRIS dataset contains 4-dimesnsional feature vectors of 150 flowers in 3 classes. Attributes are lengths and widths of petals and sepals
- GLASS dataset contains 214 objects in 7 classes. Each object has 9 features (number of components used in composition of the glass)
- MOONS is a synthetic of 500 points that form two interleaving half circles. Each point is 2-dimensional and the dataset has 2 classes

Dataset Description (Synthetic)

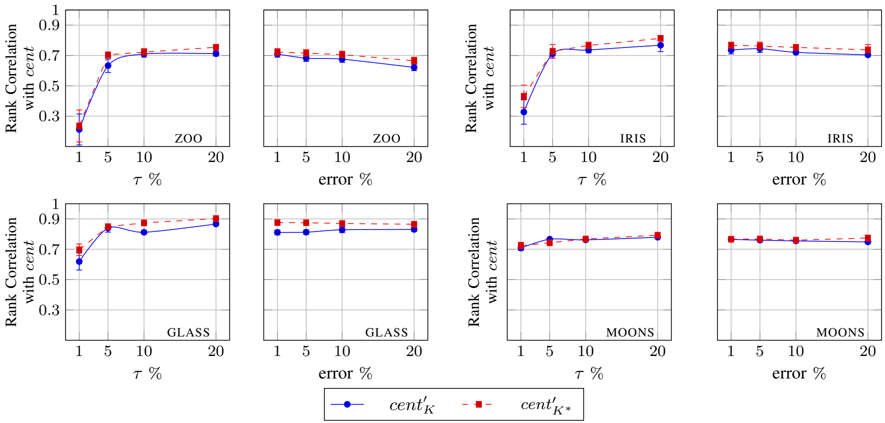
- Similarity $\mathcal{S}$ and distance matrix $\mathcal{D}$ are generated from feature vectors
- We use Euclidean similarity for IRIS, GLASS, and MOONS datasets and Cosine similarity for ZOO dataset
- We use $\mathcal{D}$ and $\mathcal{S}$ only to generate triplets and for comparison
- We randomly generate triplets by comparing the distances of two objects y and z from an anchor object x
- A triplet (x, y, z) is obtained by comparing $d(x, y)$ and $d(x, z)$ such that $d(x, y) < d(x, z)$
- We generate $\{1, 5, 10, 20\}$ % of total possible triplets and introduce *relative error* $= \{0, 1, 5, 10, 20\}$ % in generated triplets in experiments

Results (Rank Correlation with True Similarity Matrix)



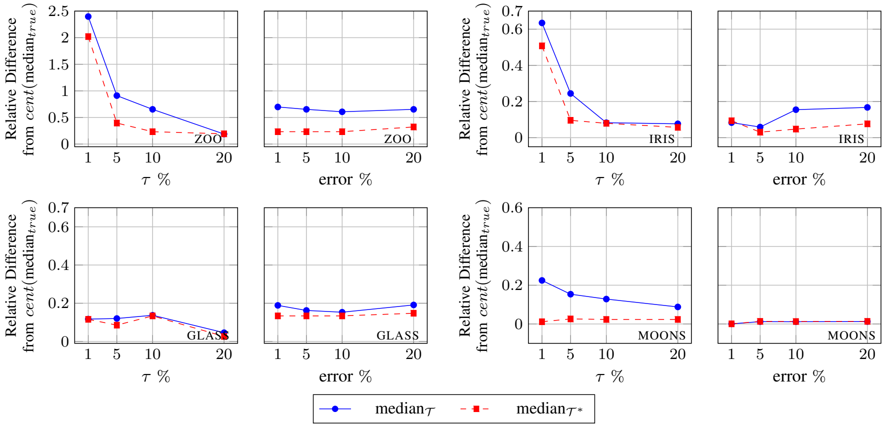
- Average row-wise rank correlation of K and K^* with $\mathcal{S}$ (true similarity matrix) for different datasets
- A higher correlation value shows more agreement with $\mathcal{S}$

Results (True vs. Approximate Centrality Vectors)



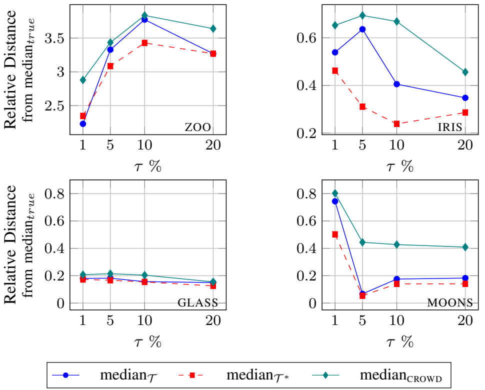
- Rank correlations of true and approximate centrality vectors
- $cent'_K$ and $cent'_{K^*}$ are centrality vectors computed from K and K^*

Results (Median Comparison)



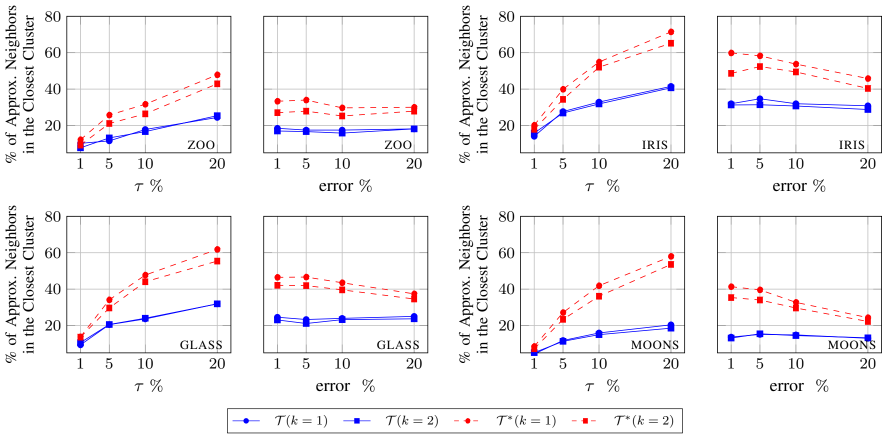
- Relative difference of $\text{median}_{\mathcal{T}}$ and $\text{median}_{\mathcal{T}^*}$ from the median_{true}
- $\text{median}_{\mathcal{T}^*}$ is generally closer to the median_{true} compared to $\text{median}_{\mathcal{T}}$

Results (Median Comparison With CROWD-MEDIAN)



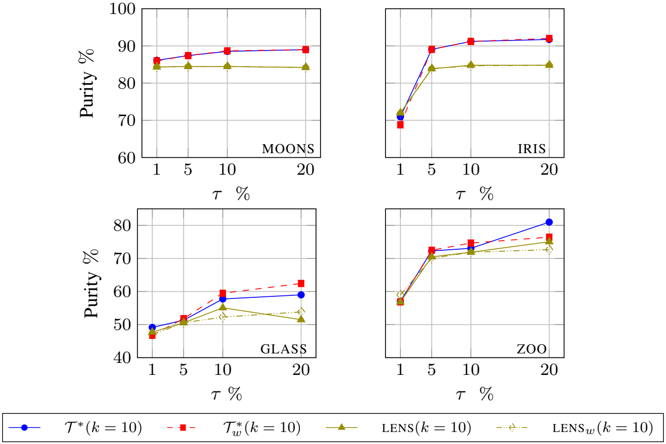
- Relative distance of CROWD-MEDIAN and ours from median_{true}
- For CROWD-MEDIAN, type $\mathbb{O}$ triplets are translated to type $\mathbb{A}$
- Our medians are closer to the median_{true} compared to $\text{median}_{\text{CROWD}}$
- τ% shows the percentage of triplets of type $\mathbb{O}$

Results (Nearest Neighbors Comparison)



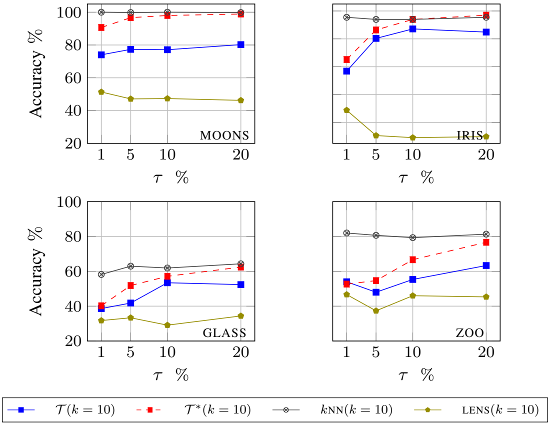
- Average percentage of approximate nearest neighbors that belong to the closest cluster of each object
- $\mathcal{T}^*(k)$ show results on augmented triplets for $k \in \{1, 2\}$ neighbors

Results (Clustering Comparison With LENSDEPTH)



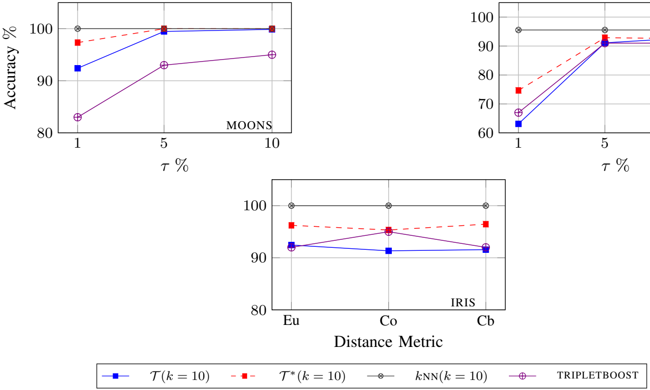
- Purity of clusterings using kNNG, k_wNNG, and LENSDEPTH ($k = 10$)
- We perform spectral clustering on kNNG and k_wNNG graphs and consider the same number of eigenvectors as of true classes

Results (Classification Comparison With LENSDEPTH)



- Classification comparison of kNN with LENSDEPTH using $\mathcal{T}$ and $\mathcal{T}^*$
- kNN shows results based on true neighbors
- In this case, τ % shows the percentage of triplets of type $\mathbb{C}$

Results (Classification Comparison With TRIPLETBOOST)



- Comparison of kNN accuracy with TRIPLETBOOST using $\mathcal{T}$ and $\mathcal{T}^*$
- The bottom figure plots results on IRIS data with τ % $= 10$ generated with Euclidean (Eu), Cosine (Co), Cityblock (Cb) distance metrics

Thank You